Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 38 (2012) 1011 – 1023

Procedia Engineering

www.elsevier.com/locate/procedia

International Conference on Modeling Optimization and Computing- (ICMOC-2012)

Wavelet Neural Network Observer Based Adaptive Tracking Control for Two Degree of Freedom Piezo- Electric Actuated Nonlinear Metal Cutting Process Using Reinforcement Learning

Manish Sharma[a], Ajay Verma[b] a*

[a]*Medicaps Institute of Technology and Management, Indore-453331, India*
[b]*Institute of Engineering and Technology, DAVV, Indore-452001, India*

Abstract

This paper is concerned with the observer designing problem for the suppression of chatter occurrence due to the existence of hysteresis and time delay in an uncertain two degree of freedom piezo-electric actuated metal cutting is proposed and the novelty is that the Wavelet Neural networks (WNN) observer using reinforcement learning is first incorporated into the controller design for a metal cutting system. An adaptive control strategy is proposed to suppress the undesirable chattering in the turning process. A piezoactuator is introduced for the regulation of the cutting tool displacement as its structure is independent of the tool holder in the machine. Reinforcement learning is used via two Wavelet Neural networks (WNN), critic WNN and action WNN, which are combined to form an adaptive WNN controller. The "strategic" utility function is approximated by the critic WNN and is minimized by the action WNN. Adaptation laws are developed for the online tuning of wavelets parameters. By Lyapunov approach, the uniformly ultimate boundedness of the closed-loop tracking error is verified. Finally, a simulation is performed to verify the effectiveness and performance of the proposed method in eliminating the chattering.

Keywords: Wavelet neural networks; chatter; metal cutting process; delayed systems; adaptive control; optimal control; reinforcement learning; nonlinear observers; Lyapunov- Krasovskii functional.

1. Introduction

A metal cutting process involves the continuous removal of material from the workpiece in the form of chips by feeding the cutting tool. However its multi-part assembly results in a complex dynamic structure, which is subjected to vibrations. These vibrations, known as chatter, must be suppressed to preserve surface finish, prevent tool and maintain the machining tolerances. There has been tremendous work done on the investigation of chatter mechanism and its suppression. Few researcher have addressed this problem in their work and proposed certain strategies like appropriate model of the structures of machine tool, fixtures, tools, choosing appropriate tool geometric angles, spindle speed control, etc.

Recently, the advent of piezoelectric devices has attracted many researcher towards the applications of piezoactuators for metal cutting process [1,2]. Piezoactuators can rapidly expand and reach the nominal displacements on the order of microsecond,

* Corresponding author. Tel.: +91-9826760499; fax: +91-0731-2856263.
E-mail address: er.mann24@gmail.com.

doi:10.1016/j.proeng.2012.06.127

with an electrical voltage applied and pushing or pulling force can be excited with the large acceleration. The implementation of piezoactuators in the metal cutting process may suppress the undesirable chatter and provides high-precision positioning control.

In order to design an accurate control schemes for the chatter suppression by using piezoactuators, the exact dynamics of the systems and the relationship between the cutting force and chip thickness has to be completely known. Many dynamic models and corresponding controllers for the metal cutting process have been presented in the literatures [1-3]. Rather than using the commonly accepted linear Merchant model, the relationship of cutting force variation and chip thickness variation is treated as a hysteresis model, which has been proved to be able to explain the nonlinearity of the turning metal cutting process in a way that is much more accurate. In this case, the turning cutting systems can be described as a class of uncertain systems with hysteresis and time delay. The relationship between the cutting force and chip thickness fluctuation is treated as hysteresis model. Hysteresis has been recognized as one of the factors that cause chatter in metal cutting process. Hysteresis is very complicated, nonsmooth nonlinearity which makes the controller design and analysis a complicated task. In addition, the effect of time delay, unavoidably resulting from the regenerative chatter effect of turning, is another major factor to be considered. Obviously, the combination of the hysteresis and time delay has proposed a challenging topic for the controller design in metal cutting process.

In metal cutting process, measurement of the states is not feasible and so the unmeasurable states are generally estimated based on the available measurement and knowledge of the physical system. Designing of a stable adaptive observer that estimates the unmeasurable states and unknown system dynamics for a class of nonlinear time delayed systems is a very special case of investigation over last few years. Most of the results cited in the literature are based on the norm based uncertainties leading to highly conservative observer design [4-7]. System identification tools like neural networks can be used to relax the conservatism up to some extent.

Recently, wavelet neural networks (WNNs), which absorbs the advantages such as the multi-resolution of wavelets and the learning of NN, were proposed to guarantee the good convergence and were used to identify and control nonlinear systems [8-10].Wavelet networks are feed-forward neural networks using wavelets as activation function. The WNN is suitable for the approximation of unknown nonlinear functions with local nonlinearities and fast variations because of its intrinsic properties of finite support and self-similarity.

Reinforcement learning (RL) is a class of algorithms for solving multi-step, sequential decision problems by finding a policy for choosing sequences of actions that optimize the sum of some performance criterion over time [11,12].In RL problems, an agent interacts with an unknown environment. At each time step, the agent observes the state, takes an action, and receives a reward. The goal of the agent is to learn a *policy* (i.e., a mapping from states to actions) that maximizes the long-term return. Actor-Critic algorithm is an implementation of RL which has separate structures for perception (critic) and action (actor) [13,14]. Given a specific state, the actor decides what action to take and the critic evaluates the outcome of the action in terms of future reward (goal).

Incorporating the advantages of WNN, adaptive actor-critic WNN-based control has emerged as a promising approach for the nonlinear systems. In the actor-critic WNN based control; a long-term as well as short-term system-performance measure can be optimized. While the role of the actor is to select actions, the role of the critic is to evaluate the performance of the actor. This evaluation is used to provide the actor with a signal that allows it to improve its performance, typically by updating its parameters along an estimate of the gradient of some measure of performance, with respect to the actor's parameters. The critic WNN approximates a certain “strategic” utility function that is similar to a standard Bellman equation, which is taken as the long-term performance measure of the system. The weights of action WNN are tuned online by both the critic WNN signal and the filtered tracking error. It minimizes the strategic utility function and uncertain system dynamic estimation errors so that the optimal control signal can be generated. This optimal action WNN control signal combined with an additional outer-loop conventional control signal is applied as the overall control input to the nonlinear system. The outer loop conventional signal allows the action and critic NNs to learn online while making the system stable. This conventional signal that uses the tracking error is viewed as the “supervisory” signal [11].

These motivate us to consider the designing of WNN observer based adaptive tracking controller for two degree of freedom metal cutting process using reinforcement learning. WNN are used for approximating the system uncertainty as well as to optimize the performance of the control strategy.

The paper is organized as follows: section II deals with the system preliminaries, wavelet observer design is given in section III. WNN based controller designing aspects are discussed in section IV. Section V describes the tuning algorithm for actor-critic wavelets. The stability analysis of the proposed control scheme and the observer is given in section VI. Effectiveness of the proposed strategy is illustrated through an example in section VII while section VIII concludes the paper.

2. System Preliminaries

2.1. Metal Cutting Process

In this paper, a two degree of freedom and two dimensional force model is used to describe the metal cutting process. The forces mentioned above are main cutting force and the thrust force as shown in figure 1. The mechanism of chatter in turning has been addressed by many researchers [1-3]. It is concluded that chatter is caused by the hysteretic relationship between the cutting force and uncut chip thickness of workpiece, which ensures the system to maintain the vibration. The hysteresis model can better explain the experimental phenomena and as well give better correlation with the theoretical analysis of dynamic cutting.

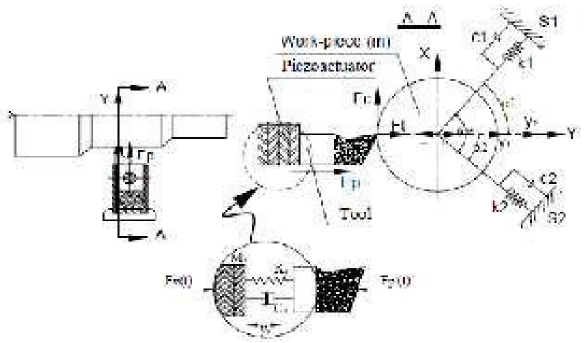

Figure 1. Schematic model of two degree of freedom metal cutting process

The dynamic equations of the system described in Fig. 1 can be presented as follows.

$$\begin{pmatrix} \frac{m}{\cos\alpha_1} & 0 \\ 0 & \frac{m}{\cos\alpha_2} \end{pmatrix} \begin{pmatrix} \frac{d^2y_1}{dt^2} \\ \frac{d^2y_2}{dt^2} \end{pmatrix} + \begin{pmatrix} \frac{\xi_1}{\cos\alpha_1} & 0 \\ 0 & \frac{\xi_2}{\cos\alpha_2} \end{pmatrix} \begin{pmatrix} \frac{dy_1}{dt} \\ \frac{dy_2}{dt} \end{pmatrix} + \begin{pmatrix} \frac{k_1}{\cos\alpha_1} & 0 \\ 0 & \frac{k_2}{\cos\alpha_2} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

$$= \begin{pmatrix} \cos\alpha_1 & \sin\alpha_1 \\ \cos\alpha_2 & -\sin\alpha_2 \end{pmatrix} \begin{pmatrix} F_t \\ F_c \end{pmatrix} + \begin{pmatrix} \cos\alpha_1 \\ \cos\alpha_2 \end{pmatrix} F_p \tag{1}$$

$$F_p = F_V - m_p \frac{d^2 y_p}{dt^2} \tag{2}$$

where $y_1(t)$ and $y_2(t)$ represents the displacements between the cutting tool and the workpiece due to each mode (S_1 and S_2) in the direction normal to the cut surface, m, ξ_i and k_i are the equivalent mass, damping constant and spring stiffness of the structure in each mode, F_t and F_c denote the main thrust force and cutting force exciting the structure, α_1 and α_2 are the geometrical angles between each mode and the direction y. F_p is the reaction force between the piezoactuator and workpiece during the cutting process, F_V is the blocked force of piezoactuator resulted from the voltage applied, y_p is the displacement between the piezoactuator and the tool holder due to the damping-spring structure existing them and m_p is the equivalent mass of the piezoactuator.

The cutting force $F(t)$ and the uncut chip thickness $u(t)$ are related as

$$F(t) = H(u(t)) \tag{3}$$

where $H(.)$ denotes the hysteresis operator.

The regenerative chatter effect in turning process results in the introduction of a time delay between successive revolutions. It means that the upper surface of the chip being removed at time t has been modulated by the cutting edge when the last cut this part of material at time $(t-\tau)$. The uncut chip thickness $u(t)$ can be described as[1]

$$u(t)=u_0(t)+y(t)-\mu y(t-\tau) \tag{4}$$

where $u_0(t)$ is the nominal uncut chip thickness, τ is the time delay between two cuts, μ is the overlapping factor due to regenerative chatter, $0<\mu\le 1$.[3]

The objective of the controller design is to maneuver the displacement $y(t)$ between the cutting tool and workpiece in turning process, so as that displacement $y(t)$ approaches to zero as $t\ge 0$. The metal cutting system finally in the state space representation may be written as

$$\begin{aligned}
\dot{x}_1 &= x_2\\
\dot{x}_2 &= \frac{\cos\alpha_1}{m}\left(a_1x_1(t)+b_1x_2(t)+c_1x_1(t-\tau)+d_1x_2(t-\tau)+e_1v(t)+\varepsilon_1\right)\\
\dot{x}_3 &= x_4\\
\dot{x}_4 &= \frac{\cos\alpha_2}{m}\left(a_2x_3(t)+b_2x_4(t)+c_2x_3(t-\tau)+d_2x_4(t-\tau)+e_2v(t)+\varepsilon_2\right)\\
y &= x_1+x_3
\end{aligned} \tag{5}$$

where

$$\begin{aligned}
a_i &= -\frac{k_i}{\cos\alpha_i}+A_t\cos\alpha_i+A_c\sin\alpha_i\\
b_i &= -\frac{\xi_i}{\cos\alpha_i}+A_t\cos\alpha_i+A_c\sin\alpha_i\\
c_i &= -\mu(\cos\alpha_i+\sin\alpha_i) \qquad\qquad i = 1 \text{ and } 2\\
d_i &= \mu(\cos\alpha_i+\sin\alpha_i)\\
e_i &= k_pP\cos\alpha_i\\
\varepsilon_i &= A_tu_0(t)\cos\alpha_i+A_cu_0(t)\sin\alpha_i\pm B_t\cos\alpha_i\pm B_c\sin\alpha_i\\
&\quad +k_pg(u(t))\cos\alpha_i-m_p\cos\alpha_i\frac{d^2y_p}{dt^2}
\end{aligned}$$

System (5) can be divided into two subsystems. The first subsystem comprises of states x_1 and x_2 and the second subsystem comprises of x_3 and x_4. Rewriting the system (5) as

$$\begin{aligned}
\dot{x}_A &= A_1x_A+B_1(f_1(x_A)+u)\\
y_A &= C_1x_A\\
\dot{x}_B &= A_2x_B+B_2(f_2(x_B)+u)\\
y_B &= C_2x_B
\end{aligned} \tag{6}$$

where $x_A=[x_1\ x_2]^T$, $x_B=[x_3\ x_4]^T$, $y=y_A+y_B$.

Also $A_i=\begin{bmatrix}0&1\\0&0\end{bmatrix}$, $B_i=\begin{bmatrix}0\\1\end{bmatrix}^T$ and $C_i=\begin{bmatrix}1\\0\end{bmatrix}^T$ $\quad i=1\ and\ 2$.

The objective is to formulate a state feedback control law to achieve the desired tracking performance. The control law is formulated using the transformed system (6). Let $\bar{y}_d = [y_d, \dot{y}_d, \ddot{y}_d, \dddot{y}_d]^T$ be the vector of desired tracking trajectory. Following assumptions are taken for the systems under consideration.

Assumption A1

a) Desired trajectory $y_d(t)$ is assumed to be smooth, continuous C^n and available for measurement.

b) For the system under consideration only output y is measurable.

2.2 Wavelet Neural Network

Wavelet network is a type of building block for function approximation. The building block is obtained by translating and dilating the mother wavelet function. Corresponding to certain countable family of a_m and b_n, wavelet function can be expressed as

$$\left\{ a_m^{-d/2} \psi\left(\frac{x - b_n}{a_m} \right) : m \in Z, n \in Z^d \right. \tag{7}$$

Considering

$$a_m = a_0^m,\ b_n = n a_0^{-m} b_0,\ m \in Z, n \in Z \tag{8}$$

The wavelet in (1) can be expressed as

$$\psi_{mn} = \left\{ a_0^{-md/2} \psi\left(a_0^{-m} x - n b_0 \right) : m \in Z, n \in Z \right\}^d \tag{9}$$

where the scalar parameters a_0 and b_0 define the step size of dilation and translation discretizations (typically a_0=2 and b_0=1) and $x = [x_1, x_2, ..., x_n]^T \in R^n$ is the input vector.

Output of an n dimensional WNN with m wavelet nodes is

$$f = \sum_{m \in Z} \sum_{n \in Z^d} \alpha_{mn} \psi_{mn} \tag{10}$$

3. Wavelet Observer Design

Wavelet based observer that estimates the states of the system (6) is given by

$$\begin{aligned} \dot{\hat{x}}_A &= A_1 \hat{x}_A + B_1(\hat{f}_1(\hat{x}_A) + u) + m_A(y_A - C_1 \hat{x}_A) \\ \hat{y}_A &= C_1 \hat{x}_A \\ \dot{\hat{x}}_B &= A_2 \hat{x}_B + B_2(\hat{f}_2(\hat{x}_B) + u) + m_B(y_B - C_2 \hat{x}_B) \\ \hat{y}_B &= C_2 \hat{x}_B \end{aligned} \tag{11}$$

where $\hat{x}_A$ and $\hat{x}_B$ are the estimations of state vectors x_A and x_B respectively, $m_A = [m_1, m_2]^T$ and $m_B = [m_1, m_2]^T$ are the observer gain matrices, selected such that the matrix $A_1 - m_A C_1$ and $A_2 - m_B C_2$ are stable. In this work WNN is used for system identification.

Assumption A2

a) $\left\| f_1(x_A(t), x_A(t-\tau)) - f_1(\hat{x}_A(t), \hat{x}_A(t-\tau)) \right\| \le \gamma_1 \left\| \tilde{x}_A \right\| + \gamma_2 \left\| \tilde{x}_A(t-\tau) \right\|$

b) $\| f_2(x_B(t), x_B(t-\tau)) - f_2(\hat{x}_B(t), \hat{x}_B(t-\tau)) \| \le \gamma_{21} \| \tilde{x}_B \| + \gamma_{22} \| \tilde{x}_B(t-\tau) \|$

c) For a symmetric positive definite matrix Q there exist a symmetric positive definite matrix P such that
$(A - mC)^T P + P(A - mC) = -Q$ and $(PB)^T = C$

where $\tilde{x} = x - \hat{x}$ is the state variable estimation error while γ_1 and γ_2 are positive constants. Defining the respective error subsystems as

$$
\begin{aligned}
\dot{\tilde{x}}_A &= (A_1 - m_A C_1)\tilde{x}_A + B_1\big(f_{1A}(x_A(t), x_A(t-\tau)) - f_{1A}(\hat{x}_A(t), \hat{x}_A(t-\tau)) + v_{r1}\big) \\
\tilde{y}_A &= C_1 \tilde{x}_A \\
\dot{\tilde{x}}_B &= (A_2 - m_B C_2)\tilde{x}_B + B_2\big(f_{2B}(x_B(t), x_B(t-\tau)) - f_{2B}(\hat{x}_B(t), \hat{x}_B(t-\tau)) + v_{r2}\big) \\
\tilde{y}_B &= C_2 \tilde{x}_B
\end{aligned}
\tag{12}
$$

The robust control terms are defined as

$$
v_{r1} = \frac{\tilde{y}_A(\rho_1^2 + 1)}{2\rho_1^2} \text{ and } v_{r2} = \frac{\tilde{y}_B(\rho_2^2 + 1)}{2\rho_2^2} \tag{13}
$$

where ρ_1 and ρ_2 are the prescribed attenuations.

4. Basic Controller Design Using Filtered Tracking Error

Defining the state tracking error vector $\hat{e}(t)$ as

$$
\hat{e}(t) = \begin{bmatrix} e_1(t) \\ \cdots\cdots \\ e_2(t) \end{bmatrix} = \begin{bmatrix} \hat{x}_1 - y_d \\ \hat{x}_2 - \dot{y}_d \\ \cdots\cdots\cdots \\ \hat{x}_3 - \ddot{y}_d \\ \hat{x}_4 - \dddot{y}_d \end{bmatrix} \tag{14}
$$

The filter tracking error is defined as

$$
\hat{r} = \hat{r}_1 + \hat{r}_2 = [k_1 \ k_2]\hat{e}_1 + [k_3 \ k_4]\hat{e}_2 \tag{15}
$$

where k_1, k_2, k_3 *and* k_4 are appropriately chosen coefficients such that $\hat{e}_1$ *and* $\hat{e}_2 \to 0$ exponentially as $\Re \to 0$.

Applying the feedback linearization method, the control law is defined as

$$
\begin{aligned}
u = \frac{m}{k_2 e_1 \cos\alpha_1 + k_4 e_2 \cos\alpha_2} \Big(& -k_1 e_2 - k_3 e_4 - k_2\big(\tfrac{\cos\alpha_1}{m}(a_1 x_1(t) + b_1 x_2(t) \\
& + c_1 x_1(t-\tau) + d_1 x_2(t-\tau) + \varepsilon_1)\big) - \dddot{y}_d - k_4\big(\tfrac{\cos\alpha_2}{m}(a_2 x_3(t) + b_2 x_4(t) \\
& + c_2 x_3(t-\tau) + d_2 x_4(t-\tau) + \varepsilon_2)\big) - \ddot{y}_d - K_r \hat{r} \Big)
\end{aligned}
\tag{16}
$$

where $K_r = [0, k_1, k_2, k_3]$.

Stability of the system (6) with the proposed control strategy will be analyzed in the subsequent section.

5. Adaptive WNN Controller Design

A novel strategic utility function is defined as the long-term performance measure for the system. It is approximated by the WNN critic signal. The action WNN signal is constructed to minimize this strategic utility function by using a quadratic optimization function. The critic WNN and action WNN weight tuning laws are derived. Stability analysis using the Lyapunov direct method is carried out for the closed-loop system (6) with novel weight tuning updates.

5.1 Strategic Utility Function

The utility function $p(k)=[p_i(k)]_{i=1}^{m}\in\Re^m$ is defined for the subsystems on the basis of the filtered tracking error $\hat{r}$ and is given by:

$$
\begin{aligned}
p_{1i}(k) &= 0 \quad \text{if } \hat{r}_{1i}^2 \le \eta_1 \\
&= 1 \quad \text{if } \hat{r}_{1i}^2 > \eta_1 \\
&\text{and} \\
p_{2i}(k) &= 0 \quad \text{if } \hat{r}_{2i}^2 \le \eta_2 \\
&= 1 \quad \text{if } \hat{r}_{2i}^2 > \eta_2
\end{aligned} \tag{17}
$$

where $p_{1i}(k)$ *and* $p_{2i}(k)\in\Re, i=1,2,\ldots,m$ and η_1 *and* $\eta_1\in\Re$ are the predefined thresholds. $p(k)$ can be considered as the current performance index defining good tracking performance for $p(k)=0$ and poor tracking performance for $p(k)=1$. The strategic utility function $Q_1'(k)\in\Re^m$ *and* $Q_2'(k)\in\Re^m$ can be defined respectively using the binary utility function as

$$
\begin{aligned}
Q_1'(k) &= \alpha_1^N p_1(k+1)+\alpha_1^{N-1} p_1(k+2)+\ldots+\alpha_1^{k+1} p_1(N)+\ \ldots \\
Q_2'(k) &= \alpha_2^N p_2(k+1)+\alpha_2^{N-1} p_2(k+2)+\ldots+\alpha_2^{k+1} p_2(N)+\ \ldots
\end{aligned} \tag{18}
$$

where $\alpha\in\Re$ and 0<α<1 and N is the horizon. Above equations may be rewritten as

$$
\begin{aligned}
Q_1(k) &= \min_{u(k)}\{\alpha_1 Q_1(k-1)-\alpha_1^{N+1} p_1(k)\} \\
Q_2(k) &= \min_{u(k)}\{\alpha_2 Q_2(k-1)-\alpha_2^{N+1} p_2(k)\}
\end{aligned} \tag{19}
$$

5.2 Critic WNN

Two critic WNNs are implemented for both the subsystems. The long term system performance can be approximated by the critic WNNs for the subsystems 1 and 2 by defining the prediction errors as

$$
\begin{aligned}
e_{1c}(k) &= \hat{Q}_1(k)-\alpha_1(\hat{Q}_1(k-1)-\alpha_1^N p_1(k)) \\
e_{2c}(k) &= \hat{Q}_2(k)-\alpha_2(\hat{Q}_2(k-1)-\alpha_2^N p_2(k))
\end{aligned} \tag{20}
$$

where $\hat{Q}_1(k)=\hat{w}_{11}^T(k)\phi_{11}(v_{11}^T x_A(k))=\hat{w}_{11}^T(k)\phi_{11}(k)$ and $\hat{Q}_2(k)=\hat{w}_{12}^T(k)\phi_{12}(v_{12}^T x_B(k))=\hat{w}_{12}^T(k)\phi_{12}(k)$, $e_{1c}(k)$ *and* $e_{2c}(k)\in\Re^m$, $\hat{Q}_1(k)$ *and* $\hat{Q}_2(k)\in\Re^m$ are the critic signals for the respective subsystems , $w_{11}(k)$ *and* $w_{12}(k)\in\Re^{n_1\times m}$ and v_1 *and* $v_2\in\Re^{nm\times n_1}$ represent the weight estimates, $\phi_{11}(k)$ *and* $\phi_{12}(k)\in\Re^{n1}$ are the wavelet activation functions and n_1 is the number of nodes in the wavelet layer. The objective functions for the two subsystems to be minimized by the critic WNN are defined as:

$$
\begin{aligned}
E_{1c}(k) &= \frac{1}{2}e_{1c}^T(k)e_{1c}(k) \\
E_{2c}(k) &= \frac{1}{2}e_{2c}^T(k)e_{2c}(k)
\end{aligned} \tag{21}
$$

The weight update rule for the critic NN is derived from gradient-based adaptation, which is given by

$$\begin{aligned}\hat{w}_{11}(k+1) &= \hat{w}_{11}(k) + \Delta\hat{w}_{11}(k) \\ \hat{w}_{12}(k+1) &= \hat{w}_{12}(k) + \Delta\hat{w}_{12}(k)\end{aligned} \tag{22}$$

where

$$\Delta\hat{w}_{11}(k) = \alpha_{11}\left[-\frac{\partial E_{1c}(k)}{\partial \hat{w}_{11}(k)}\right]$$

$$\Delta\hat{w}_{12}(k) = \alpha_{12}\left[-\frac{\partial E_{2c}(k)}{\partial \hat{w}_{12}(k)}\right]$$

or

$$\begin{aligned}&\hat{w}_{11}(k+1) = \hat{w}_{11}(k) - \alpha_{11}\phi_{11}(k) \\ &\times(\hat{w}_{11}^T(k)\phi_{11}(k) + \alpha_{11}^{N+1} p_1(k) - \alpha \hat{w}_{11}^T(k-1)\phi_{11}(k-1))^T \\ &\mathit{and} \\ &\hat{w}_{12}(k+1) = \hat{w}_{12}(k) - \alpha_{12}\phi_{12}(k) \\ &\times(\hat{w}_{12}^T(k)\phi_{12}(k) + \alpha_{12}^{N+1} p_2(k) - \alpha \hat{w}_{12}^T(k-1)\phi_{12}(k-1))^T\end{aligned} \tag{23}$$

where α_{11} *and* $\alpha_{12} \in \Re$ are the WNN adaptation gains. The critic WNNs weights are tuned by the reinforcement learning signal and the discounted past output values of critic WNN.

5.3 Action WNN

The action NN is implemented for the approximation of the unknown nonlinear functions $f_1(x_A(k))$ *and* $f_2(x_B(k))$ and to provide an optimal control signals to the overall input $u(k)$ as

$$\begin{aligned}\hat{f}_1(k) &= \hat{w}_{21}^T(k)\phi_{21}(v_{21}^T x_A(k)) = \hat{w}_{21}^T(k)\phi_{21}(k) \\ \hat{f}_2(k) &= \hat{w}_{22}^T(k)\phi_{22}(v_{22}^T x_B(k)) = \hat{w}_{22}^T(k)\phi_{22}(k)\end{aligned} \tag{24}$$

where $\hat{w}_{21}(k)$ *and* $\hat{w}_{22}(k) \in \Re^{n_2 \times m}$, v_{21} *and* $v_{22} \in \Re^{nm \times n_2}$ represent the matrices of weight estimate, $\phi_{21}(k)$ *and* $\phi_{22}(k) \in \Re^{n_2}$ are the activation function, n_2 is the number of nodes in the hidden layer. Suppose that the unknown target output-layer weights for the action WNNs are w_{21} *and* w_{22} then we have

$$\begin{aligned}f_1(k) &= w_{21}^T(k)\phi_{21}(v_{21}^T x(k)) + \varepsilon_{21}(x_A(k)) = w_{21}^T(k)\phi_{21}(k) + \varepsilon_{21}(x_A(k)) \\ f_2(k) &= w_{22}^T(k)\phi_{22}(v_{22}^T x(k)) + \varepsilon_{22}(x_B(k)) = w_{22}^T(k)\phi_{22}(k) + \varepsilon_{22}(x_B(k))\end{aligned} \tag{25}$$

where $\varepsilon_{21}(x_A(k))$ *and* $\varepsilon_{22}(x_B(k)) \in \Re^m$ are the WNN approximation errors. From (18) and (19), we get

$$\begin{aligned}\tilde{f}_1(k) &= \hat{f}_1(k) - f_1(k) = (\hat{w}_{21}(k) - w_{21})^T \phi_{21}(k) - \varepsilon_{21}(x_A(k)) \\ \tilde{f}_2(k) &= \hat{f}_2(k) - f_2(k) = (\hat{w}_{22}(k) - w_{22})^T \phi_{22}(k) - \varepsilon_{22}(x_B(k))\end{aligned} \tag{26}$$

where $\tilde{f}_1(k)$ *and* $\tilde{f}_2(k) \in \Re^m$ are the functional estimation errors for the respective subsystems. The action WNN weights are tuned by using the functional estimation errors $\tilde{f}_1(k)$ *and* $\tilde{f}_2(k)$ and the error between the desired strategic utility function $Q_d(k) \in \Re^m$ and the critic signal $\hat{Q}(k)$ for both the subsystems. Define

$$\begin{aligned}e_{1a}(k) &= \tilde{f}_1(k) + (\hat{Q}_1(k) - Q_{1d}(k)) \\ e_{2a}(k) &= \tilde{f}_2(k) + (\hat{Q}_2(k) - Q_{2d}(k))\end{aligned} \tag{27}$$

The objective is to make the utility functions $Q_{1d}(k)$ *and* $Q_{2d}(k)$ zero at every step. Thus (25) becomes

$$\begin{aligned} e_{1a}(k) &= \tilde{f}_1(k) + \hat{Q}_1(k) \\ e_{2a}(k) &= \tilde{f}_2(k) + \hat{Q}_2(k) \end{aligned} \tag{28}$$

The objective functions to be minimized by the action WNNs are given by

$$\begin{aligned} E_{1a}(k) &= \frac{1}{2} e_{1a}^T(k) e_{1a}(k) \\ E_{2a}(k) &= \frac{1}{2} e_{2a}^T(k) e_{2a}(k) \end{aligned} \tag{29}$$

The weight update rules for the action WNNs are also gradient based adaptation, which are defined as

$$\begin{aligned} \hat{w}_{21}(k+1) &= \hat{w}_{21}(k) + \Delta\hat{w}_{21}(k) \\ \hat{w}_{22}(k+1) &= \hat{w}_{22}(k) + \Delta\hat{w}_{22}(k) \end{aligned} \tag{30}$$

where

$$\Delta\hat{w}_{21}(k) = \alpha_{21}\left[-\frac{\partial E_{1a}(k)}{\partial \hat{w}_{21}(k)}\right] \text{ and } \Delta\hat{w}_{22}(k) = \alpha_{22}\left[-\frac{\partial E_{2a}(k)}{\partial \hat{w}_{22}(k)}\right]$$

or

$$\begin{aligned} \hat{w}_{21}(k+1) &= \hat{w}_{21}(k) - \alpha_{21}\phi_{21}(k)(\hat{Q}_1(k) + \tilde{f}_1(k))^T \\ \hat{w}_{22}(k+1) &= \hat{w}_{22}(k) - \alpha_{22}\phi_{22}(k)(\hat{Q}_2(k) + \tilde{f}_2(k))^T \end{aligned} \tag{31}$$

where α_{21} *and* $\alpha_{22} \in \Re$ are the WNN adaptation gains.

The WNN weight updating rule in (31) cannot be implemented in practice since the nonlinear functions $f_1(x_A(k))$ *and* $f_2(x_B(k))$ are unknown. However, using (15), the functional estimation error is given by

$$\begin{aligned} \tilde{f}_1(k) &= \dot{r}_1 - r_1 + \delta_1(k) \\ \tilde{f}_2(k) &= \dot{r}_2 - r_2 + \delta_2(k) \end{aligned} \tag{32}$$

Substituting (32) in to (31), we get

$$\begin{aligned} \hat{w}_{21}(k+1) &= \hat{w}_{21}(k) - \alpha_{21}\phi_{21}(k)(\hat{Q}_1(k) + \dot{r}_1 - r_1 + \delta_1(k))^T \\ \hat{w}_{22}(k+1) &= \hat{w}_{22}(k) - \alpha_{22}\phi_{22}(k)(\hat{Q}_2(k) + \dot{r}_2 - r_2 + \delta_2(k))^T \end{aligned}$$

To implement the weight update rules, the unknown but bounded disturbances $\delta_1(k)$ *and* $\delta_2(k)$ are taken to be zero. Then, (31) is rewritten as

$$\begin{aligned} \hat{w}_{21}(k+1) &= \hat{w}_{21}(k) - \alpha_{21}\phi_{21}(k)(\hat{Q}_1(k) + \dot{r}_1 - r_1)^T \\ \hat{w}_{22}(k+1) &= \hat{w}_{22}(k) - \alpha_{22}\phi_{22}(k)(\hat{Q}_2(k) + \dot{r}_2 - r_2)^T \end{aligned} \tag{33}$$

Coincidentally, after replacing the functional approximation error, the weight updates for the action WNNs are tuned by the critic WNNs output, current filtered tracking errors, and a conventional outer-loop signal.

6. Stability Analysis

Consider a Lyapunov-Krasovskii functional of the form

$$V=\frac{1}{2}\tilde{x}_A^T P_A \tilde{x}_A+\frac{1}{2}\tilde{x}_B^T P_B \tilde{x}_B+\frac{1}{2}\hat{r}_1^2+\frac{1}{2}\hat{r}_2^2+\int_{t-\tau}^{t} w_A(\sigma)\,d\sigma+\int_{t-\tau}^{t} w_B(\sigma)\,d\sigma$$

Differentiating it along the trajectories of the system

$$\begin{aligned}\dot{V}=&-\frac{1}{2}\tilde{x}_A^T Q_A \tilde{x}_A+\tilde{x}_A^T P_A B_1(f_1(x_A,y_d,\dot{y}_d)-\hat{f}_1(\hat{x}_A,y_d,\dot{y}_d)-v_{r1})+\tilde{x}_A^T P_A(\phi_1(x_A(t),x_A(t-\tau))-\phi_1(\hat{x}_A(t),\hat{x}_A(t-\tau)))-\frac{1}{2}\tilde{x}_B^T Q_B \tilde{x}_B\\&+\tilde{x}_B^T P_B B_2(f_2(x_B,\ddot{y}_d,\dddot{y}_d)-\hat{f}_2(\hat{x}_B,\ddot{y}_d,\dddot{y}_d)-v_{r2})+\tilde{x}_B^T P_B(\phi_2(x_B(t),x_B(t-\tau))-\phi_2(\hat{x}_B(t),\hat{x}_B(t-\tau)))\\&+\hat{r}_1(K_1\hat{e}_1+K_2\hat{e}_2+K(\phi_1(\hat{x}_A(t),\hat{x}_A(t-\tau))+m_1C_A\tilde{x}_A)+\hat{f}_1(\hat{x}_A,y_d,\dot{y}_d))+\hat{r}_2(K_3\hat{e}_3+K_4\hat{e}_4+K(\phi_2(\hat{x}_B(t),\hat{x}_B(t-\tau))+m_2C_B\tilde{x}_B)+\hat{f}_2(\hat{x}_B,\ddot{y}_d,\dddot{y}_d)\\&+u+v_{r1}+v_{r2}-\dot{y}_d-\dddot{y}_d)+w_1(x_A(t))-w_1(x_A(t-\tau))+w_2(x_B(t))-w_B(x_A(t-\tau))\end{aligned}$$

Substituting control law u in the above equation

$$\begin{aligned}=&-\frac{1}{2}Q_{1\min}\left\|\tilde{x}_A^2\right\|+\tilde{x}_A^T P_A B_1(f_1(x_A,y_d,\dot{y}_d)-\hat{f}_1(\hat{x}_A,y_d,\dot{y}_d)+\tilde{f}_1(\hat{x}_A,y_d,\dot{y}_d)-v_{r1})-\frac{1}{2}Q_{2\min}\left\|\tilde{x}_B^2\right\|+\tilde{x}_B^T\left\|P_B B_2\right\|(f_2(x_B,\ddot{y}_d,\dddot{y}_d)\\&-\hat{f}_2(\hat{x}_B,\ddot{y}_d,\dddot{y}_d)+\tilde{f}_2(\hat{x}_B,\ddot{y}_d,\dddot{y}_d)-v_{r2})+\tilde{x}_A^T P_A^T(\phi_1(x_A(t),x_A(t-\tau))-\phi_1(\hat{x}_A(t),\hat{x}_A(t-\tau)))+\hat{r}_1(v_{r1}-k_{r1}\hat{r}_1)+\tilde{x}_B^T P_B(\phi_2(x_B(t),x_B(t-\tau))\\&-\phi_2(\hat{x}_B(t),\hat{x}_B(t-\tau)))+\hat{r}_2(v_{r2}-k_{r2}\hat{r}_2)+w_1(x_A(t))-w_1(x_A(t-\tau))+w_2(x_B(t))-w_2(x_B(t-\tau))\end{aligned}$$

$$\begin{aligned}\le&-\frac{1}{2}Q_{1\min}\left\|\tilde{x}_A^2\right\|-\frac{1}{2}Q_{2\min}\left\|\tilde{x}_B^2\right\|+M_{11}\left\|\tilde{x}_A\right\|^2+M_{12}\left\|\tilde{x}_A\right\|^2+M_{13}\left\|\tilde{x}_A\right\|\left\|\tilde{x}_A(t-\tau)\right\|+M_{21}\left\|\tilde{x}_B\right\|^2+M_{22}\left\|\tilde{x}_B\right\|^2+\\&M_{23}\left\|\tilde{x}_B\right\|\left\|\tilde{x}_B(t-\tau)\right\|+\tilde{x}_A^T P_A B_1(\tilde{f}_1(\hat{x}_A,y_d,\dot{y}_d)-v_{r1})+\tilde{x}_B^T P_B B_2(\tilde{f}_2(\hat{x}_B,\ddot{y}_d,\ddot{y}_d)-v_{r2})+M_{14}+M_{24}+\hat{r}_1(-k_{r1}\hat{r}_1)\\&+\hat{r}_2(-k_{r2}\hat{r}_2)+w_1(x_A(t))-w_1(x_A(t-\tau))+w_2(x_B(t))-w_2(x_B(t-\tau))\end{aligned}$$

$$\begin{aligned}&M_{11}=\left\|P_A B_1\right\|\gamma_{13},M_{12}=P_{A\max}\gamma_{11},M_{13}=P_{A\max}\gamma_{12}\\&M_{14}=\max\left|\hat{r}_1 v_{r1}\right|+\gamma_{14},M_{21}=\left\|P_B B_2\right\|\gamma_{23},\\&M_{22}=P_{B\max}\gamma_{21},M_{23}=P_{B\max}\gamma_{22}\ \ and\ \ M_{24}=\max\left|\hat{r}_2 v_{r2}\right|+\gamma_{24}\end{aligned}$$

where $\gamma_{11},\gamma_{12},\gamma_{13},\gamma_{14},\gamma_{21},\gamma_{22},\gamma_{23},\gamma_{24}\ge 0$

Substituting the robust control term in above equation,

$$\begin{aligned}\le&-\frac{1}{2}Q_{1\min}\left\|\tilde{x}_A^2\right\|+M_{11}\left\|\tilde{x}_A\right\|^2+M_{12}\left\|\tilde{x}_A\right\|^2+M_{13}\left\|\tilde{x}_A\right\|\left\|\tilde{x}_A(t-\tau)\right\|\\&-\frac{1}{2}Q_{2\min}\left\|\tilde{x}_B^2\right\|+M_{21}\left\|\tilde{x}_B\right\|^2+M_{22}\left\|\tilde{x}_B\right\|^2+M_{23}\left\|\tilde{x}_B\right\|\left\|\tilde{x}_B(t-\tau)\right\|\\&+\frac{\rho_1^2}{2}\varepsilon_{if}^2+\frac{\rho_2^2}{2}\varepsilon_{2f}^2-\frac{\tilde{y}_A^2}{2}-\frac{\tilde{y}_B^2}{2}+M_{14}+M_{24}-k_{1r}\hat{r}_1^2+\\&w_1(x_A(t))-w_1(x_A(t-\tau))+w_2(x_B(t))-w_2(x_B(t-\tau))\end{aligned}$$

Applying Young's inequality in the fourth and eighth term of the above equation and selecting $w_1(x_A(t))=\dfrac{M_{13}}{2\mu_1^2}\left\|\tilde{x}_A\right\|^2$ *and* $w_2(x_B(t))=\dfrac{M_{23}}{2\mu_2^2}\left\|\tilde{x}_B\right\|^2$ results in

$$\le -\frac{1}{2}Q_{1\min}\|\tilde{x}_A\|^2 - \frac{1}{2}Q_{2\min}\|\tilde{x}_B\|^2 + \left(M_{11}+M_{12}+\frac{M_{13}\mu_1^2}{2}+\frac{M_{13}}{2\mu_1^2}\right)\|\tilde{x}_A\|^2$$
$$+\left(M_{21}+M_{22}+\frac{M_{23}\mu_2^2}{2}+\frac{M_{23}}{2\mu_2^2}\right)\|\tilde{x}_B\|^2+\frac{\rho_1^2}{2}\varepsilon_{1f}^2+\frac{\rho_2^2}{2}\varepsilon_{2f}^2$$
$$-\frac{\tilde{y}_A^2}{2}-\frac{\tilde{y}_B^2}{2}+M_{14}+M_{24}-k_1\hat{r}_1^2-k_2\hat{r}_2^2$$

The system is stable as long as

$$k_1\hat{r}_1^2+k_2\hat{r}_2^2+\frac{\tilde{y}_A^2}{2}+\frac{\tilde{y}_B^2}{2}+\frac{1}{2}Q_{1\min}\|\tilde{x}_A\|^2+\frac{1}{2}Q_{2\min}\|\tilde{x}_B\|^2 \ge \left(M_{11}+M_{12}+\frac{M_{13}\mu_1^2}{2}+\frac{M_{13}}{2\mu_1^2}\right)\|\tilde{x}_A\|^2$$
$$+\left(M_{21}+M_{22}+\frac{M_{23}\mu_2^2}{2}+\frac{M_{23}}{2\mu_2^2}\right)\|\tilde{x}_B\|^2+\frac{\rho_1^2}{2}\varepsilon_{1f}^2+\frac{\rho_2^2}{2}\varepsilon_{2f}^2+M_{14}+M_{24} \quad (35)$$

7. Simulation Results

Simulation is performed to verify the effectiveness of proposed reinforcement learning WNN observer based control strategy. System (6) represents the model of two degree of freedom metal cutting process of order 4. It is assumed that only output is available for measurement. The proposed observer controller strategy is applied to this system with an objective to solve the tracking problem of system. The parameters involved in the simulation example are given in table 1 and [1].The desired trajectory is taken as $y_{1d}=0.5\sin t+0.1\cos 0.5t+0.3$ for the subsystem1 and $y_{1d}=0.3\sin t+0.2\cos 0.5t+0.4$ for the subsystem 2. Initial conditions are taken as $x(0)=[0.75,0,0.5,0]^T$ .Attenuation levels for robust controller are taken as 0.01. Controller gain vector is taken as $k=[25,5,10,1]$. Wavelet networks with discrete Shannon's wavelet as the mother wavelet is used for approximating the unknown system dynamics. Wavelet parameters for these wavelet networks are tuned online using the proposed adaptation laws. Initial conditions for all the wavelet parameters are set to zero. Simulation results are shown in the figures. As observed from the figures, system response tracks the desired trajectory rapidly.

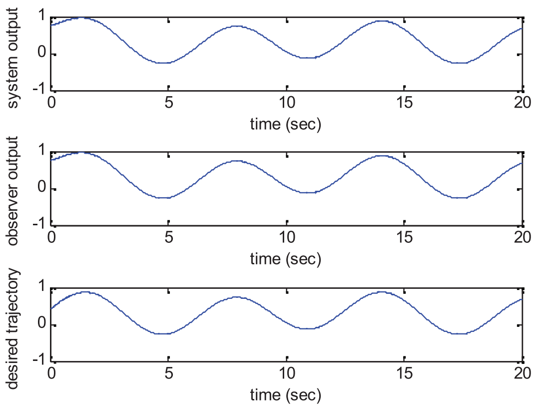

Fig. 2 System output, Observer output and desired trajectory for subsystem 1

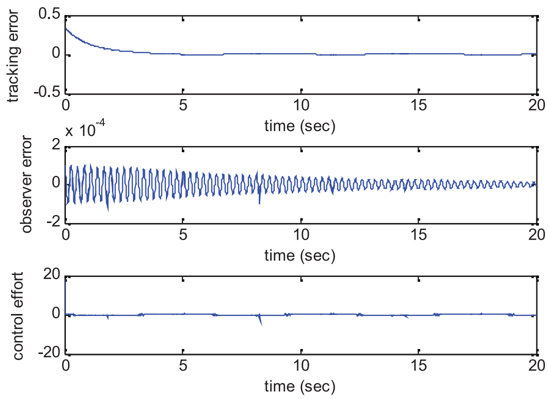

Fig. 3 Tracking error, Observer error and control effort for subsystem1

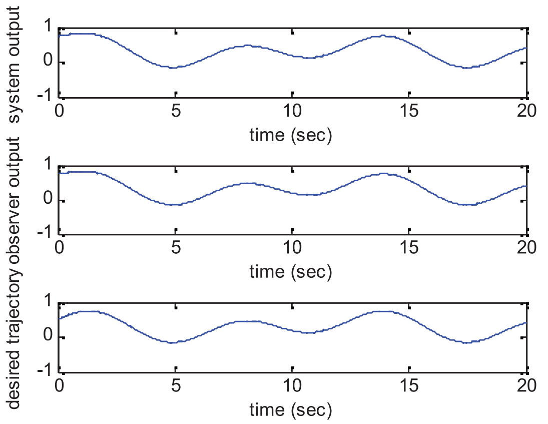

Fig. 4 System output, observer output and desired trajectory for subsystem 2

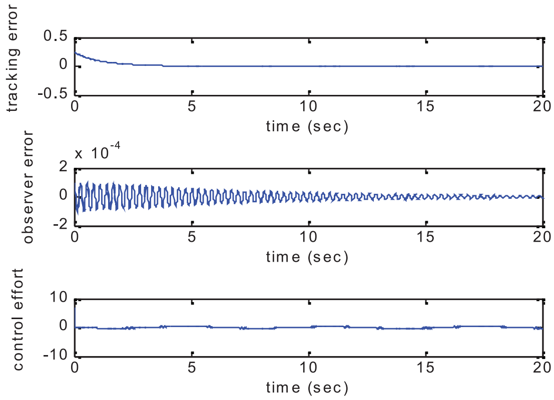

Fig. 5 Tracking error, Observer error and control effort for subsystem 2

8. Conclusion

A reinforcement learning WNN observer based adaptive tracking control strategy is proposed for two degree of freedom metal cutting process with unknown system dynamics. Adaptive wavelet networks are used for approximating the uncertainties in the system. Adaptation laws are developed for online tuning of the wavelet parameters. The stability of the overall system is guaranteed by using the Lyapunov-Krasovskii functional. The theoretical analysis is validated by the simulation results. As observed from the figure 3 and 5 that tracking errors for subsystems 1 and 2 rapidly converges to the small value of the order of 10^{-6}. A convergence pattern in the observer error is also reflected from the same figures.

Table 1. Parameters of the controller system [1]

Parameters	Values
Cutting speed	80 M/mm
Spindle speed	600 RPM
Cutting width	3 mm
Work material	S45c
Mass	100.00 Kg
c1 and c2	6320.00 & 7480.00 Ns/m
k1 and k2	4.00E+07 & 5.60E+07 N/m
α_1 and α_2	0.52 & 1.05 Radian
A_t and A_c	4.5e(+6) & 3e(+6) N/m
B_t and B_c	25.00 & 20.00 N
m_p	140 Gram
τ	2e(-3) Sec
μ	0.88
ε	1.5e(-3) mm
P	3.5e(+6)~5.5e(+6) N/m

K_p	-1.429e(+4) Lb/µm
u_0	0.15 Mm/rev

References

[1] J. Wang, C.Y.Su and M.Oya, "Theoretic modeling and adaptive control for two degree of freedom piezo-electric actuated chatter suppression system", *proceedings of the 42nd IEEE Conference on Decision and Control*, USA, pp. 4315-4320, December 2003.

[2] S. E. Oraby and A. M. Alaskari, "Mathematical Modeling Experimental Approach of the Friction on the Tool-Chip Interface of Multicoated Carbide Turning Inserts", *World Academy of Science, Engineering and Technology 75,* pp. 577-587, 2011.

[3] V.P. Astakhov, "Editorial: tribology at the forefront of study and research on metal cutting," *Int. J. Machining and Machinability of Materials*, vol. 2, no. 3, pp. 309-313, 2007.

[4] H. Lens and J. Adamy, "Observer based controller design for linear systems with input constraints", *Proceedings of 17th World Congress, The international federation of automatic control*, Seoul Korea, pp.9916-9921, July 2008.

[5] F. Abdollahi, H.A Talebi and R.V. Patel, "A stable neural network-based observer with application to flexible-joint manipulators", *IEEE Transactions on Neural Networks*, Vol. 17, Issue 1, , pp.118-129 Jan. 2006.

[6] F. Sun; Z. Sun; P. Y. Woo, "Neural network-based adaptive controller design of robotic manipulators with an observer", *IEEE Transactions on Neural Networks*, Vol. 12, Issue 1, pp.54-67 Jan. 2001.

[7] M. S. Mahmoud, Y. Shi and H. Nounou, "Resilient observer based control of uncertain time delay systems", *International journal of Innovative Computing, Information and Control*, Vol. 3 no.2, pp.407-418.Apr.2007.

[8] Q. Zhang and A. Benveniste, "Wavelet networks," *IEEE Transactions on Neural Networks*, Vol. 3, no. 6, pp.889-898, November 1992.

[9] J. Zhang, G. G. Walter, Y. Miao, and. W. Lee, "Wavelet neural networks for function learning," *IEEE Transactions on Signal Processing*, Vol. 43, no. 6, pp.1485-1497, June 1995.

[10] B. Delyon, A. Juditsky, and A. Benveniste, "Accuracy analysis for wavelet approximations," *IEEE Transactions on Neural Networks*, Vol. 6, no. 2, pp.332-348 March 1995.

[11] P. He and S. Jagannathan, "Reinforcement learning neural-network-based controller for nonlinear discrete-time systems with input constraints," *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics,* Vol. 37, no. 2, pp.425-436, April 2007.

[12] W.S. Lin, L.H. Chang and P.C. Yang, "Adaptive critic anti-slip control of wheeled autonomous robot", *IET Control Theory Applications*, Vol. 1, No. 1, January 2007.

[13] X. Lin and S. N. Balakrishnan, "Convergence analysis of adaptive critic based optimal control," *Proceedings of the American Control Conference,* pp. 1929–1933, 2000.

[14] D. V. Prokhorov and D. C. Wunsch, "Adaptive critic designs", *IEEE Transactions on Neural Networks*,, vol. 8, no. 5, pp. 997–1007, Sep. 1997.